

REMARKS

Claims 1-24 are pending in the application. Claims 8-15 stand rejected under 35 USC § 112, second paragraph, as indefinite for allegedly failing to particularly point out and distinctly claim the subject matter the Applicant regards as the invention. Claims 1-11 stand rejected under 35 USC § 102(b) as allegedly anticipated by US Patent No. 5,752,244 to Rose *et al.* (hereinafter "Rose"). Further, claims 1-11 stand rejected under 35 USC § 102(b) as allegedly anticipated by US Patent Application No. 2003/0110503 to Perkes (hereinafter "Perkes"). Claims 12-15 stand rejected under 35 USC § 103(a) as allegedly obvious over Rose *et al.* in view of Perkes. Finally, claim 22 stands rejected under 35 USC § 103(a) as allegedly obvious over Perkes. For the following reasons, the Applicant respectfully traverses.

35 USC § 112, Second Paragraph Rejection

Claims 8-15 stand rejected under 35 USC § 112, second paragraph, as being indefinite for allegedly failing to particularly point out and distinctly claim the subject matter the Applicant regards as the invention. More specifically, the Examiner asserts that the phrase "the player" in claim 8 lacks antecedent basis.

The Applicant respectfully requests that the Examiner enter the proposed amendment wherein the reference to a "media device" in the preamble has been changed to recite a "media player". Support for the amendment can be found at, *inter alia*, page 3, line 21 – page 7, line 6. As amended, the claim moots the instant rejection.

35 USC § 102e Rejections

Claims 1-11 stand rejected under 35 USC § 102(b) as allegedly anticipated by Rose et al. Further, claims 16-21, 23 and 24 stand rejected under 35 USC § 102(b) as allegedly anticipated by Perkes.

Regarding independent claim 1, the Examiner asserts that Rose et al. discloses the claimed media player wherein the storage to store media files is found in the abstract; the user interface is found in Figure 5; the content database to manage relationships between content selections and the content files is found in Figure 4; and the processor to perform at least one organization task on at least one of the content files based upon the content selection is found in column 25, lines 6-47. Applicant respectfully disagrees.

Applicant submits that Rose et al. provides a stand-alone media asset management system, i.e., a secure media storehouse, which is loosely analogous to a content source as described in the instant specification. Rose fails to provide a media player (comprising storage, user interface and processor) separate from the media source on which resides a content database to manage relationships between content selections and content files. Importantly, the instant content database is described as “allow[ing] the user to manipulate content on the player *alone*” in addition to when the player is connected to a source of media content.” Page 2, lines 15-17. As an example of media player-based control of content, the instant specification teaches that the content database allows the user to create play lists (see, e.g., page 4, lines 10-15), designate how the play lists are managed (see page 4, lines 19-23), “identify those files listed in the play list not currently resident on the player...[and] pull those files to the player storage when the player is connected to the source” (see page 4, line 25 – page 5, line 2), and even deleting media player and media source content (see, e.g., page 6, lines 5-16). As an additional indicator of the conceived expansiveness of the user database (and concomitant flexibility and power), the specification recites that “generally, the user

content database on the player will contain more media than is available on the content source.”

Page 7, lines 16-17.

In contrast, *Rose et al.* fails to provide a device remote from its media management system, let alone a remote device that allows device-based control over the media on its media management system. Similarly, *Rose et al.* fails to provide a remote media device on which its content database can describe more media than resides on the media source. Thus, *Rose et al.* fails to provide at least a media player with a content database as described in the instant invention.

Claims 2-7 depend from claim 1. As *Rose* fails to teach or suggest every element, and therefore anticipate, claim 1, it necessarily fails to anticipate claims that depend therefrom. Consequently, Applicant submits that the rejection to claims 2-7 is moot.

Regarding independent claim 8, the Examiner asserts that *Rose* discloses the claimed method. More specifically, the Examiner states that receiving a user input signal to identify a selection of content files is demonstrated by *Rose*’s browsing and searching options; that accessing a database, wherein the database provides a list of content files associated with the selection of content files is demonstrated by *Rose*’s search (and display of search results) function; and that connecting a player to a source of content and executing at least one predefined rule to perform at least one operation on at least one content file associated with the selection of content files are demonstrated by *Rose*’s provision that only authorized users may check media content in or out. Applicant respectfully disagrees.

Rose et al. describes a stand-alone, secure media storehouse, loosely analogous to a content source as described in the present specification. *Rose et al.* fails to provide any disclosure regarding a device (media player), separate from the source, that can manage relationships of the

content on and off the player, let alone a specific method to accomplish such tasks. That Rose et al. presents a method for checking media into a multimedia asset management source is immaterial because it is wholly distinct from the instant method wherein user input accesses and alters a database on a media player (the media player being separate from the one or more media sources), which database can contain information pertaining to files within and without the media player (i.e., also media on or beyond a particular media source; see, e.g., page 4, line 24 – page 5, line 2; page 7, lines 16-17). When the media player of the instant invention is again connected to a media source, the player can execute the commands elicited by the user input signal on the content stored on the player and the source.

For example, “user input from the user interface on he player...may comprise one or more of several different choices...[e.g.] the user may designate a play list...how the play list is managed when the player is connected to a remote source...[and] these signals may be stored to be operated upon when the connection is made, or some preliminary processing may be done.” Page 4, lines 16-23. Further, “when the player is connected to the content source...management tasks are performed.” Page 4, lines 25-26. Regarding the user-defined tasks that can be performed, the instant method can utilize rules such as delete all, delete none and innumerable others (e.g., page 6, lines 17 – page 7, line 22). Finally the specification illustrates one way the instant methods is distinguished from existing art:

...the player receives a user input signal 40. The player then accesses the database on the player at 50. This may or may not include identifying files that are not resident on the player. For example, the only task that the user may perform is the deletion of files. The user connects the player to the source at 42. At 60, the player executes the predetermined and selected rules to update the content on the media player. Finally, the database is updated. It must be noted that the player controls these processes, not the source of the content. For that reason, this process will be referred to as remote-directed management, to more clearly define that the media player,

remote from the source, is the controller. Page 7, line 25 – page 8, line 7;
Figure 4.

Thus, the instant method provides a way for a user to manipulate the relationships between content on a media player and one or more sources via input on the player itself, and those relationship changes can be executed once the media player is connected to a media source. After execution of the relationship changes, the media player (database) is updated. Rose *et al.* merely allows users to log media into and out of the multimedia asset management source and fails to provide methods for remote management of the content on the source (and nonexistent remote media device). Rose *et al.*'s disclosure of a search function and user's obtaining search results on its multimedia asset management source fails to teach or describe a method for a remote device receiving user input changing the relationship of content on the remote device and one or more content sources and, upon connecting to a source, the remote media device controlling the execution of the relationship changes. Further, Rose *et al.*'s disclosure of users assigned to a project checking media into and out of its multimedia asset management source fails to teach or suggest a method whereby a remote device (media player), onto which user input has defined changes in the status of the media content on the player and source, controls the execution of media content status changes on the remote device and a source. Thus, Rose *et al.* fails to disclose all elements of the instant claim and, as such, fails to anticipate the claim.

Claims 9-11 depend from claim 8. As Rose also fails to teach or suggest every element of claim 8, it necessarily fails to anticipate claims that dependent on claim 8. Consequently, Applicant submits that the rejection to claims 9-11 is moot.

Claims 16-21, 23 and 24 stand rejected under 35 USC § 102(b) as allegedly anticipated by Perkes. Regarding independent claim 16, the Examiner asserts that Perkes discloses the claimed

method of adding content on a media player. More specifically, the Examiner states that receiving a user input signal at the player, wherein the user input signal identifies a selection of content files is met by Perkes's description of a user requesting multimedia; accessing a database, wherein the database provides a list of content files associated with the collection of content files is met by Perkes's description that its method searches for requested content; determining if any content files in the list of content files do not exist on the player, connecting the player to a source of content, and adding any content files not already exist on the media player are met by Perkes's allowing users to purchase media objects. Regarding the last elements of instant claim 16, the Examiner concludes that "clearly the method has to determine if any content is not already existing on the media player and connect the player to a source of content before purchasing. Applicant respectfully disagrees.

Perkes describes a media-on-demand (MOD) system. In contrast to the present invention, however, Perkes fails to teach or suggest a method for adding content to a media player that comprises a database, which media player can function remote from the media sources. Moreover, the instant database, through user input on the article, allows the user to store changes in the relationship between media content on the article and content on the one or more media sources. When connected to a media source, the article, rather than the source, controls the execution of the changes in media relationship according to user-selected, pre-defined rules. That Perkes's method searches for requested data fails to teach or suggest accessing a database on an article that can be remote from media sources, which then controls the updating of the relationships between the media on the article and the one or more sources. Likewise, that Perkes allows for purchase of media fails to teach or suggest an article whereby user input defines changes in the status of the media content on the player and one or more sources and the article (and database thereon) controls the execution

of media content status changes on the remote device and the one or more media sources. Thus, Perkes fails to disclose all elements of the instant claim and, as such, fails to anticipate the claim.

Claims 17-18 depend from claim 16. Perkes fails to teach or suggest every element of claim 16 and consequently fails to anticipate claims that depend from claim 16. As such, the rejection to claims 17-18 is moot.

Regarding independent claim 19, the Examiner asserts that Perkes teaches the claimed article. Applicant respectfully disagrees.

Perkes describes a media-on-demand (MOD) system. In contrast to the present invention, Perkes fails to teach or suggest an article, remote from the one or more sources of content, that comprises a database. That database that, through user input on the article, allows the user to store changes in the relationship between media content on the article and content on the one or more media sources. When connected to a media source, the article, rather than the source, controls the execution of the changes in media relationship according to user-selected, pre-defined rules. That Perkes's article searches for requested data fails to teach or suggest accessing a database on the article, which article can be remote from media sources, wherein the article and its database then control the updating of the relationships between the media on the article and the one or more sources. Likewise, that Perkes allows for purchase of media fails to teach or suggest an article whereby user input defines changes in the status of the media content on the player and one or more sources and the article (and database thereon) controls the execution of media content status changes on the remote device and the one or more media sources. Thus, Perkes fails to disclose all elements of claim 19 and, as such, fails to anticipate the claim.

Claims 20-21, 23 and 24 all depend from claim 19. Perkes fails to teach or suggest all elements of independent claim 19 and thus fails to anticipate its dependent claims. Consequently, the rejection against claims 20-21, 23 and 24 is moot.

35 USC § 103a Rejections

Claims 12-15 stand rejected under 35 USC § 103(a) as allegedly obvious over Rose *et al.* in view of Perkes.

Claims 12-15 depend from independent claim 8. As argued above, Rose fails to anticipate claim 8. Thus for the combination of Rose and Perkes to render obvious claims that depend from claim 8, Perkes must remedy the deficiencies of Rose with respect to claim 8. However, Perkes also fails to teach or suggest a method for updating content on a media player as discussed in relation to claims 16 and 19 above. Because Perkes fails to provide the elements missing from Rose *et al.*, it necessarily fails to render obvious claims 12-15 in combination with Rose *et al.*

Claim 22 stands rejected under 35 USC § 103(a) as allegedly obvious over Perkes. However, claim 22 depends from independent claim 19, which, as argued above, Perkes fails to anticipate. Because Perkes fails to teach every element of claim 19, it necessarily fails to provide those same base elements relied upon to allegedly render obvious the limitation introduced in claim 22. Consequently, Applicant respectfully requests that the rejection be withdrawn.

Claim Objections

According to the Examiner's suggestion, Applicant has amended claim 8 to recite "connecting" instead of "connection of" at line 5.

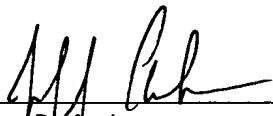
CONCLUSION

Reconsideration of this application is respectfully requested and a favorable determination is earnestly solicited. The Patent Office is invited to contact the undersigned representative if it is believed that this would be helpful in expediting prosecution of this application. The Applicant submits that the pending claims are in condition for allowance, and issuance of a Notice of Allowance is respectfully requested.

If the Examiner is of the opinion that a telephone conference would expedite prosecution of the application, the Examiner is encouraged to contact Applicant's undersigned representative.

Respectfully submitted,

Date: April 14, 2004

By: 
Jeffrey D. Anderson
Reg. No. 51,403

McDonnell Boehnen Hulbert & Berghoff LLP
300 South Wacker Drive
Chicago, Illinois 60606-6709
312-913-0001
anderson@mbhb.com